A SYSTEM FOR AUTOMATIC DETECTION OF PARTIALLY OCCLUDED OBJECTS FROM REAL-WORLD IMAGES

Predrag Neskovic*, Liang Wu and Leon N Cooper

Department of Physics and
Institute for Brain and Neural Systems
Brown University, Providence, RI 02912

ABSTRACT

In this work we consider the Bayesian Integrate And Shift (BIAS) model for learning object categories and test its performance on learning and recognizing different object categories from real-world images. In contrast to conventional learning algorithms that require hundreds or thousands of training examples, we show that our system can learn a new object category from only a few examples. In addition, our system provides information not only about the object category but also about the local regions within the object on which it is fixating. We tested the performance of the system on very challenging examples of partially occluded targets. The training was done on different instances of one category and tested on partially occluded examples that the system had never seen before. We demonstrate that the system is very robust to partial occlusions and clutter and can recognize a target even if it fixates on the occluded part.

1. INTRODUCTION

Detection and identification of partially occluded targets in complex scenes becomes an increasingly important task in light of the latest developments in urban warfare. The construction of a system that can automatically identify selected targets or direct soldiers attention to the locations that may contain suspicious activity can be of great use not only as a tool that can reduce the cognitive workload of the soldier but also as a tool that can alert the soldier to possible threats.

Identifying a target in a complex scene is a challenging problem that incorporates several important aspects of vision including: translation and scale invariant recognition, robustness to noise and ability to cope with significant variations in lighting conditions. Identifying an occluded target adds another layer of complexity and this problem can be extremely difficult even for humans. Motion information can be of great help in providing an initial figure-ground segmentation. However, in many situations motion information is not available. In addition, if the input to the system is a video stream then the requirement that the system works in real-time often precludes the use of more sophisticated but computationally involved techniques.

One of the main limitations of classical vision algorithms, such as those utilizing Artificial Neural Networks (ANNs), Radial Basis Functions (RBFs), and Support Vector Machines (SVMs), is that they require a fixed size input. This means that during the recognition phase the input vector to the system has to be of the same size as the input vector used during the training process. Such systems are therefore not well suited for occlusion problems where sections of the input vector are simply missing or carry incorrect information.

In addition, supplying a fixed size input to the recognition system requires the selection of the specific region from the image. This means that such systems have to solve the segmentation problem, find the boundary of the region occupied by the target. However, given an image, it is not known where the target is or what its size is. In order to detect a target, regardless of its location, the detection system is usually (as presented in Schneiderman and Kanade (2000) convolved over the whole image and in order to detect a target at different scales the original image is rescaled and the convolution procedure repeated. Since the methods that rely on exhaustive search are not computationally efficient, they are mostly applied to detection of targets in static images.

Human visual system, on the other hand, does not require any "presegmentation" of the image in order to recognize a specific object. In fact, when we look at an object, our visual system processes not only information coming from the object itself but the whole scene. This is accomplished through an array of neurons that are selective to specific features and whose receptive fields (RFs) are spatially distributed and localized. Although our visual system processes information from all the regions of the scene, it appears as if it somehow knows to "discard" certain regions (the background) and integrate only information from the object regions. If we are not able to recognize an object from a single fixation, then we make saccades, combine evidence from different fixations and as a result usually improve our perception of the object.

Since our visual system integrates information from neurons that have localized receptive fields, it seems natural to

Report Documentation Page

Form Approved
OMB No. 0704-0188

1. REPORT DATE	2. REPORT TYPE	3. DATES COVERED
01 NOV 2006	**N/A**	**-**

4. TITLE AND SUBTITLE	5a. CONTRACT NUMBER
A System For Automatic Detection Of Partially Occluded Objects From Real-World Images	5b. GRANT NUMBER
	5c. PROGRAM ELEMENT NUMBER
6. AUTHOR(S)	5d. PROJECT NUMBER
	5e. TASK NUMBER
	5f. WORK UNIT NUMBER
7. PERFORMING ORGANIZATION NAME(S) AND ADDRESS(ES) **Department of Physics and Institute for Brain and Neural Systems Brown University, Providence, RI 02912**	8. PERFORMING ORGANIZATION REPORT NUMBER
9. SPONSORING/MONITORING AGENCY NAME(S) AND ADDRESS(ES)	10. SPONSOR/MONITOR'S ACRONYM(S)
	11. SPONSOR/MONITOR'S REPORT NUMBER(S)

12. DISTRIBUTION/AVAILABILITY STATEMENT
Approved for public release, distribution unlimited

13. SUPPLEMENTARY NOTES
See also ADM002075., The original document contains color images.

14. ABSTRACT

15. SUBJECT TERMS

16. SECURITY CLASSIFICATION OF:			17. LIMITATION OF ABSTRACT	18. NUMBER OF PAGES	19a. NAME OF RESPONSIBLE PERSON
a. REPORT	b. ABSTRACT	c. THIS PAGE			
unclassified	**unclassified**	**unclassified**	**UU**	**8**	

represent an object as a collection of localized features. In contrast to *global* models, such as those that use a Principal Components Analysis (PCA) approach, feature-based approaches are much more robust to partial occlusions. Over the past years, feature-based approaches had become increasingly popular within the computer vision community Lowe (1999); Schmid and Mohr (1997); Serre et al. (2005); Heisele et al. (2001); Torralba et al. (2004). These approaches have been succesfully used in various applications such as face recognition Schneiderman and Kanade (2000); Viola and Jones (2001), handwriting recognition Wang et al. (2005); Neskovic et al. (2000), car detection Agarwal et al. (2004); Schneiderman and Kanade (2000); Neskovic et al. (2004), and modeling human bodies Felzenszwalb and Huttenlocher (2005). One of the problems of probabilistic feature based approaches (such as Fei-Fei et al. (2003)) is that they can not model an object with a large number of features since calculating the joint probabilities would require an enormous amount of training data. Another problem is how to find the best constellation of features. In one-dimensional case this problem can be solved using a dynamic programming approach but for two dimensional case this is still an open problem and no exact solution that is at the same time computationally efficient exists today. In contrast to approaches presented in Fei-Fei et al. (2003); Serre et al. (2005), our model uses much simpler features and does not require a feature learning stage. Furthermore, unlike the model of Fei-Fei *et al.*, our system can use an arbitrarily large number of features without an increase in computational complexity.

The main question therefore is how to deal with computational complexity when analyzing large amounts of information contained in visual scenes. It seems natural, that in designing a system for scene analysis we should use some properties of the best existing system for analyzing visual scenes - the human visual system. Unfortunately, biologically inspired models Keller et al. (1999); Rybak et al. (1998) and models of biological vision Amit and Mascaro (2003); Mel (1997); Riesenhuber and Poggio (1999) have been much less successful (in terms of real-world applications) compared to computer vision approaches. A model that captures some properties of human saccadic behavior and represents an object as a fixed sequence of fixations has been proposed by Keller *et al.* Keller et al. (1999). Similarly, Hecht-Nielsen and Zhou Hecht-Nielsen. and Zhou (1995) and Rybak *et al.* Rybak et al. (1998) presented models that are inspired by the scanpath theory Noton and Stark (1971). Although these models utilize many behavioral, psychological and anatomical concepts such as separate processing and representation of "what" (object features) and "where" (spatial features: elementary eye movements) information, they still assume that an object is represented as a sequence of eye movements. In contrast to these approaches, our model does not assume any specific sequence of saccades and therefore is more general.

In this work we consider the Bayesian Integrate And Shift (BIAS) Neskovic et al. (2006) model for learning object categories and test its performance on learning and recognizing different object categories from real-world images. In contrast to conventional learning algorithms, such as ANNs, that require hundreds or thousands of training examples, we show that our system can learn a new object category from only a few examples. In addition, our system provides information not only about the object category but also about the local regions within the object on which it is fixating. We tested the performance of the system on very challenging examples of partially occluded targets. The training was done on different instances of one category and tested on partially occluded examples that the system had never seen before. We demonstrate that the system is very robust to occlusions and clutter and can recognize a target even if it fixates on the occluded part.

The paper is organized as follows. In section 2 we give an overview of the BIAS model for learning new object categories. In section 3 we discuss implementation details. In section 4 we illustrate the performance of the system when tested on different object categories and several instances of occluded faces. In section 5 we summarize the main properties of our model and the impact of the system on the warfighter.

2. THE MODEL

Our model falls into a category of feature-based approaches Fei-Fei et al. (2003); Lowe (1999); Torralba et al. (2004); Serre et al. (2005); Schneiderman and Kanade (2000); Viola and Jones (2001). The problem that we want to solve is as follows: given a collection of features, their locations $\vec{X}$, and appearances $\vec{A}$ we want to calculate the probability that they represent an object of a specific class n, $P(O^n|\vec{X},\vec{A})$. Since calculating this probability is extremely difficult if the number of features is large, we seek to find suitable approximations. One of the biggest simplifications is to assume that the feature locations are fixed and that all the variations are due to appearances. Unfortunately, this is one of the least reasonable assumptions which holds in only few practical situations.

In order to make the model more realistic, one should include tolerance to variations in feature locations. Instead of assuming that a feature is located at a point, we will assume that it is located within a region. The question is how to design these regions? If we use large regions, we can then easily capture all possible variations in feature locations (excellent generalization) but at the expense of losing location specificity which would decrease discrimination capability of the model. On the other hand, very small regions would provide excellent localization but would lead to poor generalization. We propose that the solution to this trade-off between generalization and retaining location specificity is to use retina-like distribution of regions

in combination with saccade-like shifts. If we want to estimate the location of a specific feature, then the size of the region where it can be found (the uncertainty) depends on the location of the point with respect to which we measure its distance - the center. The further away the feature is from that center, the larger the uncertainty. Therefore, in order to capture variations in feature locations, the sizes of the regions, as well as their overlaps, have to increase with their distance from the center. As a consequence, the accuracy of estimating feature locations is high only for the features that are close to the center. In order to obtain good location estimates for the features that are further away from the center, the recognition system would have to shift the center, to make a "saccade".

Modeling an object. Let us now assume that we are given a large number of regions that form a fixed grid and completely cover an input image. Each such region we call a receptive field (RF) and with it we associate a group of feature detectors that signal the presence of the features to which they are selective. This fixed mask of the RFs can be positioned anywhere in the image and the location over which the smallest RF is positioned is the fixation point. We will call a configuration consisting of the outputs of feature detectors associated with a specific fixation point a *view*. Since there can be as many views as there are (fixation) points within the object, it means that that the number of views can be extremely large even for objects of small sizes. In order to reduce the number of views, we will assume that some views are sufficiently similar to one another so that they can be clustered into the same view. In this way, an object is modeled as a collection of views and therefore has as many labels as there are views.

Notations. With symbol H we denote a random variable with values $H = (n, i)$ where n goes through all possible object classes and i goes through all possible views within the object. Instead of (n, i), we use the symbol H_i^n to denote the i^{th} view of an object of the n^{th} (object) class. The background class, by definition, has only one view. With variable $\vec{y}$ we measure the distances of the centers of the RFs from the fixation point. The symbol D_k^r denotes a random variable that takes values from a feature detector that is positioned within the RF centered at $\vec{y}_k$ from the central location, and is selective to the feature of the r^{th} (feature) class, $D_k^r = d^r(\vec{y}_k)$. The symbol A_t denotes the outputs of all the feature detectors for a given fixation point $\vec{x}_t$ at time t. With variable $\vec{z}$ we measure the distances of the previous fixation locations (view centers) with respect to the location of the current fixation point. For example, the symbol z_{t-1}^j denotes the location of the center of the j^{th} view at time $t-1$. The collection of the locations of all the view centers, up to time t, we denote with the symbol B_t.

What we want to calculate is how spatial information, coming from different feature detectors, as well as information from previous fixations (the centers of the previous views) influence our hypothesis, $p(H_i^n|A_t, B_t)$. In order to gain a better insight into dependence of these influences, we will start by including the evidence coming from one feature detector and then increase the number of feature detectors and fixation locations.

Combining information within a fixation. Let us now assume that for a given fixation point $\vec{x}_0$, the feature of the r^{th} class is detected with confidence $d^r(\vec{y}_k)$ within the RF centered at $\vec{y}_k$. The influence of this information on our hypothesis, H_i^n, can be calculated using the Bayesian rule as

$$p(H_i^n|d^r(\vec{y}_k), \vec{x}_0) = \frac{p(d^r(\vec{y}_k)|H_i^n, \vec{x}_0)p(H_i^n|\vec{x}_0)}{p(d^r(\vec{y}_k)|\vec{x}_0)}, \quad (1)$$

where the normalization term indicates how likely it is that the same output of the feature detector can be obtained (or "generated") under any hypothesis, $p(d^r(\vec{y}_k)|\vec{x}_0) = \sum_{n,i} p(d^r(\vec{y}_k)|H_i^n, \vec{x}_0)p(H_i^n|\vec{x}_0)$.

We will now assume that a feature detector with RF centered around $\vec{y}_q$ and selective to the feature of the p^{th} class outputs the value $d^p(\vec{y}_q)$. The influence of this new evidence on the hypothesis can be written as

$$p(H_i^n|d^p(\vec{y}_q), d^r(\vec{y}_k), \vec{x}_0) =$$

$$\frac{p(d^p(\vec{y}_q)|d^r(\vec{y}_k), H_i^n, \vec{x}_0)p(H_i^n|d^r(\vec{y}_k), \vec{x}_0)}{p(d^p(\vec{y}_q)|d^r(\vec{y}_k), \vec{x}_0)}. \quad (2)$$

The main question is how to calculate the likelihood term $p(d^p(\vec{y}_q)|d^r(\vec{y}_k), H_i^n, \vec{x}_0)$? In principle, if the pattern does not represent any object but just a random background image the outputs of the feature detectors $d^p(\vec{y}_q)$ and $d^r(\vec{y}_k)$ are independent of each other. If, on the other hand, the pattern represents a specific object, say an object of the n^{th} class, then the local regions of the pattern within the detectors RFs, and therefore the features that capture the properties of those regions, are not independent from each other, $p(d^p(\vec{y}_q)|d^r(\vec{y}_k), H^n, \vec{x}_0) \neq p(d^p(\vec{y}_q)|H^n, \vec{x}_0)$. However, once we introduce a hypothesis of a specific view, the features become much less dependent on one another. This is because the hypothesis H_i^n is much more restrictive and at the same time more informative than the hypothesis about only the object class, H^n. Given the hypothesis H^n, each feature depends both on the locations of other features and the confidences with which they are detected (outputs of feature detectors). The hypothesis H_i^n significantly reduces the dependence on the locations of other features since it provides information about the exact location of each feature *within* the object up to the uncertainty given by the size of the feature's RF.

The likelihood term, under the independence assumption, can therefore be written as $p(d^p(\vec{y}_q)|d^r(\vec{y}_k), H_i^n, \vec{x}_0) = p(d^p(\vec{y}_q)|H_i^n, \vec{x}_0)$. Note that this property is very important from a computational point of view and allows for a very fast training procedure. The dependence of the hypothesis

on the collection of outputs of feature detectors A_0 can be written as

$$p(H_i^n|, A_0, \vec{x}_0) =$$

$$\frac{\prod_{rk\in A} p(d^r(\vec{y}_k)|H_i^n, \vec{x}_0)p(H_i^n|\vec{x}_0)}{\sum_{n,i}\prod_{rk\in A} p(d^r(\vec{y}_k)|H_i^n, \vec{x}_0)p(H_i^n|\vec{x}_0)} \quad (3)$$

where r, k goes over all possible feature detector outputs contained in the set A_0 and n, i goes over all possible hypotheses.

Combining information across fixations. We now calculate how the evidence about the locations of different fixations influence the confidence about the specific hypothesis, H_j^n, associated with fixation point $\vec{x}_t$. We assume that at time $t-1$ a hypothesis has been made that the fixation at distance $\vec{z}_{t-1}^{\,i}$ from the current fixation represented the center of the i^{th} view of the object of the n^{th} class. Similarly, we will assume that at time $t-2$ a hypothesis has been made that the fixation at distance $\vec{z}_{t-2}^{\,k}$ from the current fixation represented the center of the k^{th} view. We denote with the symbol A_t the outputs of all the feature detectors that are used to calculate the (new) hypothesis H_j^n. The influence of the evidence about the locations of the previous hypotheses on the current hypothesis can be written as

$$p(H_j^n|\vec{z}_{t-1}^{\,k}, \vec{z}_{t-2}^{\,i}, A_t, \vec{x}_t) = \quad (4)$$

$$\frac{p(\vec{z}_{t-1}^{\,k}|H_j^n, \vec{z}_{t-2}^{\,i}, A_t, \vec{x}_t)p(H_j^n|\vec{z}_{t-2}^{\,i}, A_t, \vec{x}_t)}{p(\vec{z}_{t-1}^{\,k}|\vec{z}_{t-2}^{\,i}, A_t, \vec{x}_t)}.$$

In order to make the model computationally tractable, we will assume that the view locations are independent from one another given the hypothesis.

Since the location of the k^{th} view of the object does not depend on the configuration of feature detectors that is associated with the current view, and assuming that view locations are independent from one another, the likelihood term from Eq. (4) becomes $p(\vec{z}_{t-1}^{\,k}|H_j^n, \vec{z}_{t-2}^{\,i}, A_t, \vec{x}_t) = p(\vec{z}_{t-1}^{\,k}|H_j^n, \vec{x}_t)$. The probability that the input pattern represents the j^{th} view of the object of the n^{th} class, given the activations of the letter detectors A_t and locations of other views B_t can be written as

$$p(H_j^n|A_t, \vec{x}_t, B_t, f(s)) =$$

$$\frac{\prod_{s<t} p(\vec{z}_s^{\,f(s)}|H_j^n, \vec{x}_t)p(H_j^n|A_t, \vec{x}_t)}{\sum_i \prod_{s<t} p(\vec{z}_s^{\,f(s)}|H_j^n, \vec{x}_t)p(H_j^n|A_t, \vec{x}_t)} \quad (5)$$

where i goes through views of the n^{th} object, s goes through the locations of all the fixations and the function $f(s)$ maps a location $\vec{y}_s$ to a specific hypothesis. With symbol B_t we denoted the set of the locations of all the fixations (object views) with respect to the location of the current fixation, $\vec{x}_t$. The second term in the numerator is calculated using Eq. (3).

3. IMPLEMENTATION

Modeling Likelihoods. We model the likelihoods in Eq. (3) using Gaussian distributions. The probability that the output of the feature detector representing the feature of the r^{th} class and positioned within the RF centered at $\vec{y}_k$ has a value $d^r(\vec{y}_k)$, given a specific hypothesis and the location of the fixation point, is calculated as

$$p(d^r(\vec{y}_k)|H_i^n, \vec{x}_t) = \frac{1}{\sigma_k^r\sqrt{2\pi}} exp\frac{-(\mu_k^r - d^r(\vec{y}_k))^2}{2(\sigma_k^r)^2}. \quad (6)$$

This notation for the mean and the variance assumes a particular hypothesis so we omitted some indices, $\sigma_k^r = \sigma_k^r(n, i)$. The values for the mean and variance are calculated in the batch mode but, as we will see in the next section, only a small number of instances are used for training so the memory requirement is minimal. For modeling the location likelihoods in Eq. (5) we use the multivariate Gaussian distributions since in this case the mean location is a vector and similarly the variance is a covariance matrix. Note also the difference in measuring the location of the center of a specific RF, $\vec{y}_k$, and in measuring the location of the fixation point $\vec{z}^{\,k}$. Although both distances are calculated with respect to the same reference point (the fixation point) the locations of the RFs form a fixed grid while the locations of fixation points can vary continuously.

Feature Detectors and Receptive Fields. In this work we extract features using a collection of Gabor filters where a Gabor function that we use is described with the following equation

$$\psi_{f_0,\theta,\sigma}(x, y) = \frac{e^{-\frac{1}{8\sigma^2}(4(xcos\theta+ysin\theta)^2+(ycos\theta-xsin\theta)^2)}}{\sqrt{2\pi}\sigma}\cdot$$

$$sin(2\pi f_0(xcos\theta + ysin\theta)). \quad (7)$$

The inspiration for selecting these features comes from the fact that simple cells in the visual cortex can be modeled by Gabor functions as shown by Marcelja Marcelja (1980) and Daugman Daugman (1980).

One way to constrain the values of the free parameters in Eq. (7) is to use information from neurophysiological data on simple cells as suggested by Lee Lee (1996). More specifically, the relation between the spatial frequency and the bandwidth can be derived to be: $2\pi f_o\sigma = 2\sqrt{ln2}(2^\phi + 1)/(2^\phi - 1)$ (see Lee (1996) for more detail). Since the spatial frequency bandwidths of the simple and complex cells have been found to range from 0.5 to 2.5 octaves, clustering around 1.2 octaves, we set ϕ to 1.5 octaves. The orientations and bandwidths of the filters are set to: $\theta = \{0, \pi/4, \pi/2, 3\pi/4\}$ and $\sigma = \{2, 4, 6, 8\}$. Each RF has a square form and the size of the smallest RF is 31x31 pixels. The RFs are arranged along 8 directions and the sizes of the RFs are increased at the ratio of 1.4 (controlled by the enlarge parameter). For example, the sizes of the RFs that are

nearest neighbors to the central RF are (31x1.4)x(31x1.4). The overlap between two neighboring receptive fields is 50% meaning that for two neighboring RFs, the larger RF covers 50% of the area of the smaller receptive field. The recognition results are not very sensitive to the small changes in the overlap, enlarge parameter, and the sizes of the receptive fields.

With each RF we associate 16 feature detectors where each feature detector signals the presence of a feature (i.e. a Gabor filter of specific orientation and size) to which it is selective no matter where the feature is within its receptive field. One way to implement this functionality is to use a max operator. The processing is done in the following way. On each region of the image, covered by a specific RF, we apply a collection of 16 Gabor filters (4 orientations and 4 sizes) and obtain 16 maps. Each map is then supplied to a corresponding feature detector and the feature detector then finds a maximum over all possible locations. As a result, each feature detector finds the strongest feature (to which it is selective) within its RF but does not provide any information about the location of that feature. This makes the number of features that our system uses over 1,000.

The Training Procedure. The training is done in a supervised way. We constructed an interactive environment that allows the user to mark a section of an object and label it as a fixation region associated with a specific view. Therefore, every point within this region can serve as the view center. Once the user marks a specific region, the system samples the points within it and calculates the mean and variance for each feature detector. Since the number of training examples is small the training is very fast. Note that during the training procedure the input to the system is the whole image and the system learns to discriminate between an object and the background. It is important to stress that the system does not learn parts of the object, but the whole object from the perspective of the specific fixation point.

4. RESULTS

We tested the performance of our system on four object categories (faces, cars, airplanes and motorcycles) using the Caltech database as in Fei-Fei et al. (2003); Serre et al. (2005). As a performance measure we used the error rate at equilibrium point (EP), which is calculated by setting the threshold so that the miss rate is equal to the false positive rate. We chose this measure over the Receiver Operator Characteristic (ROC) since it provides more compact representation of the results, in the sense that much more information can be represented in one graph compared to ROC measure. For illustrative purposes, we chose the face category to present some of the properties of our system in more detail.

The system was first trained on background images in order to learn the “background” hypothesis. We used 20

Fig. 1. View regions as selected by the teacher.

random images and within each image the system made fixations at 100 random locations. The system was then trained on specific views of specific objects. For example, in training the system to learn the face from the perspective of the right eye, the user marks with the cursor the region around the right eye and the system then makes fixations within this region in order to learn it.

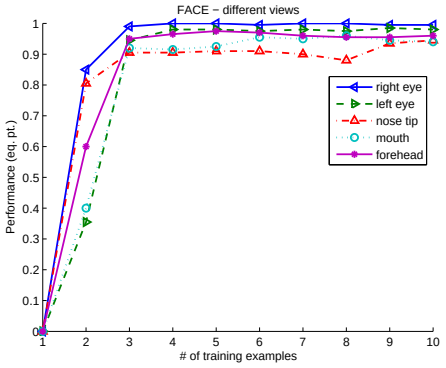


Fig. 2. Performance graphs for different views of the face. The task is to verify whether an image contains a face and to estimate the locations of different views within the face.

During the testing phase, the system makes random fixations and for each fixation point we calculated the probability that the configuration of the outputs of feature detectors represents a face from the perspective of the right eye. To make sure that there are also positive examples among the random fixations, each testing image is divided into the view region(s) (in this case the right eye region) and the rest of the image represents the ”background” class. Therefore, positive examples consisted of random fixation within the region of the right eye and negative examples consisted of random fixations outside the region of the right eye. The

Table 1.

Faces - Multi Views				
	1	**2**	**3**	**4**
r. eye	98.0 %	99.0 %	99.5 %	100 %
l. eye	94.5 %	99.5 %	100 %	100 %
nose	90.5 %	94.0 %	96.5 %	97.5%
mouth	92.0 %	97.5 %	98.5 %	99.0 %

Table 2.

Cars - Multi Views				
	1	**2**	**3**	**4**
v1	88.2 %	91.1 %	94.2%	95.8 %
v2	86.6 %	91.5%	93.3 %	94.0 %
v3	88.9 %	90.4 %	93.0 %	94.9 %
v4	84.4 %	90.7 %	92.5%	93.2 %

system was tested on instances that were not used for training. We used 200 positive examples and 1000 negative examples for testing. In all of the experiments that follow, we set the number of fixations per view (the number of sampling points) to 10.

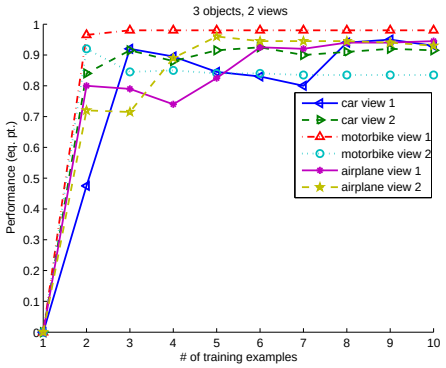
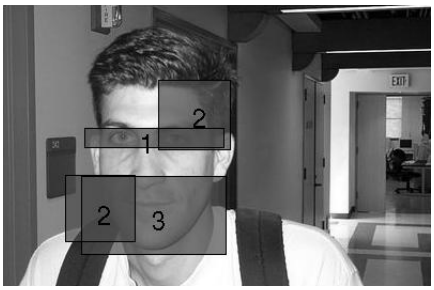


Fig. 3. Performance graphs for three different objects using two different views. The task is to recognize a specific object category and estimate locations of different views withing the object.

As illustrated in Figure 2, the system can easily learn a new view from only a few training examples. Since the system was not able to learn much from one example, we set the performance to zero for one training example. In order to learn the face (and "discard" information from the background) it has to be presented with more than one training example. As it turns out, two examples are not quite enough, as can be clearly seen in Figure 2, but with three examples the system can learn the new face (the specific view of the face) with high confidence.

In Figure 3 we show that the system can easily learn classes other than faces. For each class we used two views



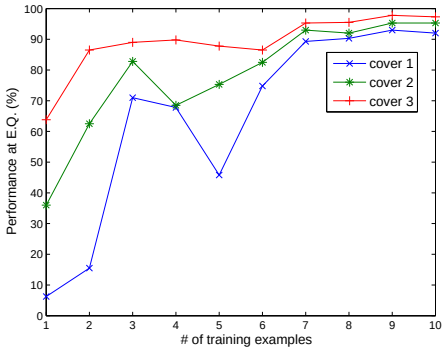


Fig. 4. Top: Three different occlusions used for testing the system on the face it has never seen before. The task is to detect a face and estimate the location of the right eye. Bottom: Corresponding performances under different occlusions.

as illustrated in Figure 1. The system was tested on instances it has never seen before. The task was to detect an object within the image and estimate the locations of the specific views.

Although the performance of the system is very good using only a single view, we tested whether and how much information from other fixations improves the performance, Tables 1 and 2. The tests were done on faces and cars and we used 4 very good views for faces and 4 below the average views for cars. In both cases the information about the spatial location of other views improved the performance. During the training phase, the user marks the fixation (view) regions and the system then calculates the location likelihoods for each pair of regions separately by randomly selecting n points from each region. During the testing phase, in order to estimate the location of the view center, the system selects 10 points with the highest probabilities (as representing the view) and takes the average over their locations. This location is then selected by the system as a representing the view center.

The system was first trained on individual views and the results are illustrated in column **1** of Tables 1 and 2. When the system used information about the location of one more view, the performance improved, as shown in column **2**.

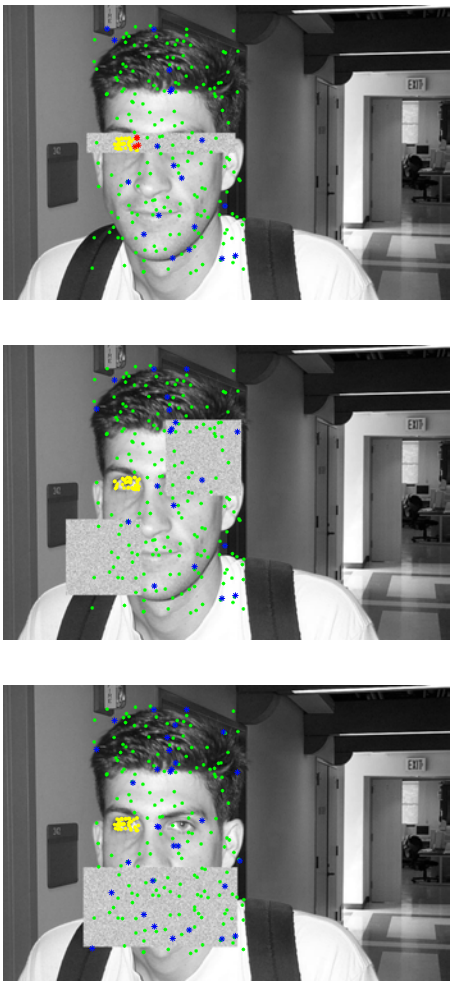


Fig. 5. Performance of the system on different face occlusions. Yellow stars denote correctly detected positive fixations, green stars denote correctly detected negative fixations, red stars denote missed fixations, and blue stars denote false alarm fixations.

Note that this information is not provided by the teacher but estimated by the system. This means that the recognition rates can decrease if the system erroneously estimates the view centers. However, utilizing information about the location of two different views always improved the performance as captured by the numbers in column **3**. The best performance, as expected, was obtained when the system used the information about centers of the three views as shown in column **4**.

Since our system uses information from over 1,000 feature detectors distributed over the whole image, it is very robust to occlusions. This is demonstrated in Figure 4, bottom, that illustrates the performance of the system when tested on occluded images as shown in Figure 4, top . The system was trained to recognize a face category from the perspective of the right eye (righ-eye-view) using (non-occluded) examples from different people.

The system was tested on three types of occlusions: a) the bar covering both eyes (denoted as cover 1 in Figure 4, b) two large disconnected regions covering the face (cover 2), and c) the rectangle covering the face below the nose (cover 3). Tests were done on face images of people that were not used for training. As one can see, system can recognize the face even when the fixating region is covered (Figure 5, top), which means that it utilizes information from the whole face and not only local information around the fixation point. We use yellow stars to display correctly detected positive fixations, green stars for correctly detected negative fixations, red stars for missed fixations, and blue stars for false alarm fixations, Figure 5. Incorrect fixations are bigger in size.

5. CONCLUSIONS

In this work we considered the Bayesian Integrate And Shift (BIAS) Neskovic et al. (2006) model for learning object categories and tested its performance on learning and recognizing different object categories from real-world images. In contrast to conventional learning algorithms, such as ANNs, that require hundreds or thousands of training examples, we showed that our system can learn a new object category from only a few examples. In addition, our system provides information not only about the object category but also about the local regions within the object on which it is fixating.

We tested the performance of the system on very challenging examples of partially occluded targets. The training was done on different instances of one category and tested on partially occluded examples that the system had never seen before. We demonstrated that the system is very robust to partial occlusions and clutter and can recognize a target even if it fixates on the occluded part.

The benefit of this system to the soldier will be twofold: it will reduce the cognitive workload of the soldier operating in complex visual environments (such as those encountered in urban combat), and it will alert the soldier to possible threats that might otherwise be overlooked due to the partial occlusions. We believe that the system for automatic detection of concealed and partially occluded target will have a significant impact on the warfighter especially in light of the latest developments in urban warfare.

ACKNOWLEDGEMENTS

This work is supported in part by ARO under grant W911NF-04-1-0357.

REFERENCES

Agarwal, S., A. Awan, and D. Roth, 2004: Learning to detect objects in images via a sparse, part-based representation. *PAMI*, **26**, 1475–1490.

Amit, Y. and M. Mascaro, 2003: An integrated network for invariant visual detection and recognition. *Vision Research*, **43**, 2073–2088.

Daugman, J. G., 1980: Two-dimensional spectral analysis of cortical receptive field profile. *Vision Research*, **20**, 847–856.

Fei-Fei, L., R. Fergus, and P. Perona, 2003: A bayesian approach to unsupervised one-shot learning of object categories. *Proc. ICCV*.

Felzenszwalb, P. and D. Huttenlocher, 2005: Pictorial structures for object recognition. *Intl. Journal of Computer Vision*, **61**, 55–79.

Hecht-Nielsen., R. and Y. Zhou, 1995: VARTAC: A foveal active vision ATR system. *Neural Networks*, **8**, 1309–1321.

Heisele, B., T. Serre, M. Pontil, T. Vetter, and T. Poggio, 2001: Categorization by learning and combining object parts. *Proc. NIPS*.

Keller, J., S. Rogers, M. Kabrisky, and M. Oxley, 1999: Object recognition based on human saccadic behaviour. *Pattern Analysis and Applications*, **2**, 251–263.

Lee, T. S., 1996: Image representation using 2d gabor wavelets. *PAMI*, **18**, 1–13.

Lowe, D., 1999: Object recognition from local scale-invariant features. *Proc. ICCV*.

Marcelja, S., 1980: Mathematical description of the responses of simple cortical cells. *J. Optical Soc. Am.*, **70**, 1,297–1,300.

Mel, B., 1997: Seemore: Combining color, shape and texture histogramming in a neurally-inspired approach to visual object recognition. *Neural Computation*, **9**, 777–804.

Neskovic, P., P. Davis, and L. Cooper, 2000: Interactive parts model: an application to recognition of on-line cursive script. *Advances in Neural Information Processing Systems*, 974–980.

Neskovic, P., D. Schuster, and L. Cooper, 2004: Biologically inspired recognition system for car detection from real-time video streams. *Neural Information Processing: Research and Development*, J. C. Rajapakse and L. Wang (Eds.), Springer - Verlag, 320–334.

Neskovic, P., L. Wu, and L. Cooper, 2006: Learning by integrating information within and across fixations. *Proc. ICANN*.

Noton, D. and L. Stark, 1971: Scanpaths in eye movements during pattern perception. *Science*, **171**, 308–311.

Riesenhuber, M. and T. Poggio, 1999: Hierarchical models of object recognition in cortex. *Nat. Neurosci.*, **2**, 1019–1025.

Rybak, I. A., V. I. Gusakova, A. Golovan, L. N. Podladchikova, and N. A. Shevtsova, 1998: A model of attention-guided visual perception and recognition. *Vision Research*, **38**, 2387–2400.

Schmid, C. and R. Mohr, 1997: Local greyvalue invariants for image retrieval. *PAMI*, **19**, 530–534.

Schneiderman, H. and T. Kanade, 2000: A statistical method for 3d object detection applied to faces and cars. *Proc. CVPR*.

Serre, T., L. Wolf, and T. Poggio, 2005: Object recognition with features inspired by visual cortex. *Proc. CVPR*.

Torralba, A., K. P. Murphy, and W. T. Freeman, 2004: Sharing features: efficient boosting procedures for multiclass object detection. *Proc. CVPR*.

Viola, P. and M. Jones, 2001: Rapid object detection using a boosted cascade of simple features. *Proc. CVPR*.

Wang, J., P. Neskovic, and L. Cooper, 2005: A probabilistic model for cursive handwriting recognition using spatial context. *Proc. ICASSP*.